

Document made available under the Patent Cooperation Treaty (PCT)

International application number: PCT/EP05/000862

International filing date: 28 January 2005 (28.01.2005)

Document type: Certified copy of priority document

Document details: Country/Office: DE
Number: 10 2004 004 345.0
Filing date: 29 January 2004 (29.01.2004)

Date of receipt at the International Bureau: 30 March 2005 (30.03.2005)

Remark: Priority document submitted or transmitted to the International Bureau in compliance with Rule 17.1(a) or (b)



World Intellectual Property Organization (WIPO) - Geneva, Switzerland
Organisation Mondiale de la Propriété Intellectuelle (OMPI) - Genève, Suisse

BUNDESREPUBLIK DEUTSCHLAND

15.03.05

**Prioritätsbescheinigung über die Einreichung
einer Patentanmeldung**

Aktenzeichen: 10 2004 004 345.0

Anmeldetag: 29. Januar 2004

Anmelder/Inhaber: ABB Research Ltd., Zürich/CH

Bezeichnung: System und Verfahren zur Kommunikation zwischen
entfernten Objekten und lokalen Stellvertretern

IPC: H 04 L 12/16

Die angehefteten Stücke sind eine richtige und genaue Wiedergabe der ursprünglichen Unterlagen dieser Patentanmeldung.

München, den 14. März 2005
Deutsches Patent- und Markenamt
Der Präsident
Im Auftrag

Walther

ABB Research Ltd.

Zürich

Mp.-Nr. 04/511

27. Januar 2004

PAT 3-Sf/Bk

System und Verfahren zur Kommunikation zwischen entfernten Objekten und lokalen Stellvertretern

Beschreibung

Die Erfindung bezieht sich auf ein System und ein Verfahren zur Kommunikation zwischen entfernten Objekten, deren Methoden als XML-Web-Services – hier auch kurz als Webservices bezeichnet - zugreifbar sind, und ihren lokalen Stellvertreterobjekten, deren Methoden einen transparenten Zugriff auf die entfernten Methoden implementieren. Neben der Bezeichnung Stellvertreter wird hier auch die Bezeichnung Proxy verwendet. Solche Kommunikationsverfahren sind beispielsweise in <http://www.w3c.org/2002/ws/>, (Stand: 27.01.2004), beschrieben.

XML-Web-Services stellen eine standardisierte und weit verbreitete Grundlage zur Kommunikation in lose gekoppelten, verteilten Systemen, z. B. im Internet, insbesondere zum Aufruf entfernter Prozeduren und Methoden dar. In objektorientierten Systemen können XML-Web-Services zum Zugriff auf entfernte Objekte eingesetzt werden.

In Fig. 3 ist die Systemarchitektur eines Kommunikationssystems nach dem Stand der Technik dargestellt. Dabei ist ein Klient oder Client 1 dargestellt, der mit einem Dienstanbieter 10 über das Internet oder ein LAN 9 kommuniziert. Der Client 1 ist als Datenverarbeitungseinrichtung mit zugehöriger Software aufzufassen, mit wenigstens einer Client-Anwendung 2, die mit Proxies 3, z. B. Proxy A und Proxy B kommuniziert, wobei diese wiederum mit einer Kommunikationsschicht 4 kommunizieren. Die Kommunikationsschicht 4 des Client 1 ist dafür eingerichtet, mit

einer entsprechenden Kommunikationsschicht 4 eines Diensteanbieters 10 über das Internet 9 zu kommunizieren. Die Einrichtungen des Diensteanbieters 10 enthalten außerdem Software-Mittel zur Realisierung von Services 5, z. B. Service A und Service B.

Der bekannte Ablauf eines entfernten Dienstaufrufs in einer solchen Umgebung gemäß Fig.3 ist nachstehend beschrieben:

- i. Der Client 1 ruft eine lokale, von einem Stellvertreter 3 zur Verfügung gestellte Prozedur auf.
- ii. Der Stellvertreter 3 verwendet die in der Kommunikationsschicht 4 implementierte Funktionalität zum Übermitteln der Prozeduraufrufinformationen über das LAN oder das Internet 9 zum Diensteanbieter 10.
- iii. In der Kommunikationsschicht 4 des Diensteanbieters 10 wird die Prozeduraufrufinformation extrahiert, die den Dienst implementierende Prozedur 5 lokal aufgerufen, der Rückgabewert des Aufrufs formatiert und an die Kommunikationsschicht 4 des Klienten 1 übermittelt.
- iv. Die Kommunikationsschicht 4 des Klienten 1 extrahiert die Prozedurrückgabeinformationen, und der entsprechende lokale Stellvertreter 3 liefert den Rückgabewert als Ergebnis des initialen lokalen Prozeduraufrufs an die Client-Anwendung 2.

Durch die Verwendung von SOAP (Simple Object Access Protocol), wie in <http://www.w3.org/TR/soap12-part1/> beschrieben, verbunden mit http (Hypertext Transfer Protocol), wie in <ftp://ftp.isi.edu/in-notes/rfc2616.txt> beschrieben, als Verfahren zum Austausch strukturierter und typisierter Daten, können Dienste unterschiedlichsten Anwendungen innerhalb des sogenannten WWW über Plattformgrenzen hinweg zugänglich gemacht werden. Grundsätzlich ist das bekannte, wie auch das unten beschriebene Verfahren jedoch unabhängig vom verwendeten Protokoll.

Ein wesentlicher Nachteil einer solchen generischen Realisierung entfernter Methodenaufrufe ist, dass diese Aufrufe um ein Vielfaches zeit- und rechenintensiver

sind als lokale Funktionsaufrufe oder Funktionsaufrufe in enger gekoppelten Systemen.

Der Erfindung liegt daher die Aufgabe zugrunde, ein System und ein Verfahren zur Kommunikation zwischen entfernten Objekten anzugeben, mit denen eine Verringerung des Kommunikationsaufwands erreichbar ist.

Diese Aufgabe wird durch ein Kommunikationssystem gelöst, das die im Anspruch 1 angegebenen Merkmale aufweist. Ein zugehöriges Kommunikationsverfahren und vorteilhafte Ausgestaltungen sind in weiteren Ansprüchen angegeben.

Mit der Erfindung wird demnach vorgeschlagen, den erforderlichen Kommunikationsaufwand dadurch zu verringern, dass Client-seitig eine Optimierungsschicht sowie ein Generalproxy angeordnet werden, und beim Dienstanbieter einen Generalservice einzufügen. Mit diesen Maßnahmen lassen sich Optimierungen, z. B. durch Bündelung von Aufrufen erreichen.

Das System bzw. Verfahren erlaubt die Nutzung von Zwischenspeichern (Caches), alternativen Datenformaten und Übertragungsprotokollen, sowie anwendungsspezifischen Optimierungen.

In Umgebungen, die z. B. über die Grenzen eines lokalen Netzwerkes hinweg einzelne Komponenten nur lose koppeln, gilt insbesondere, dass ein Bündeln von Dienstaufrufen wünschenswert sein kann, wenn die fixen Kosten für eine sogenannte Rundreise vom Klienten zum Dienstanbieter und zurück im Verhältnis zu den Kosten der Übertragung und Bearbeitung der eigentlichen Dienstanfrage und Antwort sehr hoch sind. Diese Kosten werden verursacht durch (oft mehrfach) notwendige Verbindungsaufbauten, die Erstellung und Übertragung protokollspezifischer Nachrichtenköpfe und die Initialisierung von Nachrichten verarbeitenden (z. B. von SOAP/XML-Parser-, Authentifizierungs- und Autorisierungs-) oder übertragenden (z. B. HTTP-Verbindungs- oder Datenkomprimierungs-) Komponenten. Mit der Erfindung wird hinsichtlich dieses Sachverhalts eine Verbesserung erreicht, da ermöglicht wird, unabhängig von der Erstellung der Dienste (Webservices) und Stellvertreter (Proxies), Optimierungen in den Kommunikationsablauf einzufügen. Dabei werden

keine zusätzlichen Anforderungen an die Ablaufumgebung sowohl der Dienst- als auch der Klient- und Stellvertreterkomponenten gestellt, und die generischen Dienst- und Stellvertreterkomponenten, insbesondere deren Schnittstellen, werden nicht verändert, wodurch vorteilhaft Wiederverwendbarkeit und leichte Konfigurierbarkeit erreicht werden.

Das erfindungsgemäße Verfahren ermöglicht es:

1. entfernte Methodenaufrufe zu unterbinden, z. B. indem lokale Caches zum Beantworten von Informationsanfragen genutzt werden,
2. entfernte Methodenaufrufe zu verzögern, z. B. können die in objektorientierten Umgebungen notwendigen Methodenaufrufe zur Kontrolle der Lebenszeit entfernter Objekte, sei es durch deren explizite Zerstörung (sogenannte Destruktoraufrufe), oder durch das periodische Ermitteln von nicht mehr verwendeten Objekten (die sogenannte Garbage Collection), durchaus zeitlich verzögert vom Klienten zur Ablaufumgebung der entfernten Objekte übermittelt werden,
3. entfernte Methodenaufrufe zu bündeln, z. B. um die Zahl ressourcenintensiver Verbindungsaufbauvorgänge zu verkleinern und die in 1. und 2. hintangestellten Anfragen bei sich bietender Gelegenheit nachträglich zu tätigen,
4. die die Rückgabewerte entfernter Methodenaufrufe enthaltenden Nachrichten mit weiteren, für die lokale Anwendung relevanten Informationen anzureichern, z. B. um das Löschen von ungültig gewordenen Einträgen in lokalen Caches zu veranlassen,
5. über verwendete Übertragungsprotokolle oder Datenformate je nach Laufzeitumgebung zu entscheiden, z.B. um die ressourcenintensive SOAP/HTTP-Übertragung durch eine effizientere zu ersetzen, sofern die Art der Verbindung (Internet/LAN, eventuell vorhandene HTTP-Proxies und Firewalls) dies erlauben, und
6. zur Verwaltung, insbesondere Aktualisierung und Invalidierung der Daten in einem Zwischenspeicher, unabhängig von Aufrufen aus Client-Anwendungen eine Kommunikation zu initiieren, oder huckepack Informationen zusammen mit der Übertragung von Aufrufbündeln und der Rückübertragung von Antworten beim Dienstanbieter anzufordern.

Das Verfahren ist in allen Umgebungen einsetzbar, in denen generische Dienstanbieter- und Stellvertreter-Komponenten zur Verfügung stehen, die die Funktionalität zur Formatierung und Übertragung eines entfernten Dienstaufrufs und der entsprechenden Antwort kapseln. Zum Beispiel bietet die Framework Class Library (FCL) der Microsoft.NET Plattform, vergl. J. Richter, Applied Microsoft .NET Framework Programming, Microsoft Press 2001, Seiten 21 bis 24, solche Komponenten an.

Eine weitere Beschreibung der Erfindung und deren Vorteile erfolgt nachstehend anhand eines in Zeichnungsfiguren dargestellten Ausführungsbeispiels.

Es zeigt:

- Fig. 1 die Systemarchitektur eines erfindungsgemäßen Kommunikationssystems,
- Fig. 2 einen erfindungsgemäßen Kommunikationsablauf,
- Fig. 3 die Systemarchitektur eines Kommunikationssystems nach dem Stand der Technik.

Fig.1 zeigt die Systemarchitektur eines erfindungsgemäßen Kommunikationssystems. Zum Zweck der Kontrolle und Optimierung der Kommunikation ist dabei auf der Seite eines Dienstanbieters 10 zusätzlich zu den in einer Ausführungsumgebung gemäß dem Stand der Technik, wie bereits oben anhand der Fig. 3 beschrieben, vorhandenen Diensten, den Services 5, ein weiterer Webservice allgemeinerer Funktionalität installiert, der mit Generalservice 8 bezeichnet ist. Dieser ist in der Lage, eine oder mehrere Dienstanfragen an die ursprünglichen Dienstanbieter, die Services 5, zu vermitteln, und eine oder mehrere Antwortnachrichten an den oder die entfernten Klienten, die Clients 1, zu übermitteln.

Entsprechend ist in der entfernten Ablaufumgebung, beim Client 1, ein generischer Stellvertreter für diesen zusätzlichen Webservice installiert, der als Generalproxy 7 bezeichnet ist. Außerdem enthält der Client 1 als zusätzliche Komponente eine Optimierungsschicht 6. Diese bietet für die Kommunikation mit den Proxies 3 erforderliche Kommunikations- und Datenformatierungs-Funktionen und ersetzt insoweit Funktionen der Kommunikationsschicht 4, erweitert aber auch die

Optimierungsmöglichkeiten. Die Optimierungsschicht 6 kann ihrerseits mittels des Generalproxies 7 die Funktionalität der Kommunikationsschicht 4, oder eine andere Implementierung der benötigten Funktionalität, verwenden.

Weitere Informationen zur Funktion der bereits genannten Komponenten erschließen sich aus der nachstehenden Beschreibung eines Kommunikationsablaufes anhand der Fig. 2. In Fig. 2 symbolisieren senkrechte Balken die jeweils benutzten Systemkomponenten, deren Nummerierung mit der jeweiligen Komponentenbezeichnung in Fig. 3 übereinstimmt. Mit S1 bis S14 sind Ablaufschritte bezeichnet, die nachstehend erläutert werden.

Fig. 2 zeigt einen typischen Ablauf eines Dienstaufwurfes in einer erfindungsgemäß erweiterten Umgebung:

Schritt S1: Der Client 1, bzw. die Client-Anwendung 2 ruft eine lokale, von einem Proxy 3 zur Verfügung gestellte Prozedur auf.

Schritt S2: Der Proxy 3 verwendet die in der Optimierungsschicht 6 implementierte Funktionalität zum Übermitteln von Prozeduraufrufen. Die Optimierungsschicht 6 bietet dem Proxy oder Stellvertreter 3 dazu eine zur ursprünglichen Kommunikationsschicht identische Schnittstelle.

Schritt S3: In der Optimierungsschicht 6 wird der Dienstaufwurf, sofern er nicht verzögert oder durch ein, in einem Zwischenspeicher der Optimierungsschicht 6 vorhandenes Ergebnis beantwortet werden kann, durch schon vorhandene, vorher verzögerte, oder aber sinnvollerweise im Voraus zu tätige Dienstaufwürfe, erweitert (Bündelung von Aufrufen).

Schritt S4: Dieses Bündel von Aufrufen wird durch einen lokalen Aufruf des Generalproxy 7 an die generische Kommunikationsschicht 4 des Client 1 übergeben, und zur Kommunikationsschicht 4 des Dienstansbieters 10 übertragen.

Schritt S5: Durch einen lokalen Aufruf der Generalserviceprozedur wird das Bündel an den Generalservice 8 übergeben.

Schritt S6: Die Generalserviceprozedur bearbeitet die Dienstaufwürfe und liefert deren Ergebnisse. Der Generalservice 8 findet dazu im Schritt 6 die benötigten Services 5 auf.

Schritt S7: Methodenaufwurf durch den Generalservice 8.

Schritt S8: Resultate-Rückgabe der Services 5 an den Generalservice 8.

Schritt S9: Zusammenstellung der Resultate durch den Generalservice 8 zu Bündeln.

Schritt S10: Rückgabe des Resultate-Bündels an die Kommunikationsschicht 4. Das Bündel kann erweitert sein um zusätzlich vom Dienstanbieter zur Optimierungsschicht 6 des Klienten zu übermittelnde Informationen.

Schritt S11: Übertragung zur entfernten Kommunikationsschicht 4 und von dort Rückgabe zum Generalproxy 7, der das Ergebnis als lokale Prozedurrückgabe an die Optimierungsschicht 6 liefert.

Schritt S12: Auswertung der Resultate in der Optimierungsschicht 6.

Schritt S13: In der Optimierungsschicht 6 werden die zusätzlich zur Antwort auf die initiale Dienstanfrage gelieferten Informationen ausgewertet und verwendet.

Schritt S14: Die Antwort auf die initiale Anfrage wird über den generischen Stellvertreter 3 an die Clientanwendung 2 übermittelt, womit der beispielhafte Ablauf beendet ist.

In einer konkreten Implementierung des Systems, zum Beispiel als Komponente innerhalb des .NET Frameworks, kann die Erweiterung eines bestehenden Systems von generischen Stellvertretern und Dienstanbietern durch einen Austausch der klientenseitigen Kommunikationsschicht durch die erfindungsgemäße Optimierungsschicht 6 mit Kommunikationsschicht 4 und die Installation des zusätzlichen Generalproxy parallel zu den vorhandenen Proxies realisiert werden. Anwendungsunabhängige Optimierungen, wie die Verwendung von Caches und die Verzögerung von Dienstaufrufen können so ohne Implementationsaufwand konfigurierbar zur Verfügung gestellt werden. Anwendungsabhängige Optimierungen können als Module an einer definierten Schnittstelle der Optimierungsschicht 6 eingefügt werden.

Entsprechend wird auf der Dienstanbieterseite der Generalservice 8 parallel zu den vorhandenen Services realisiert. Der Generalservice 8 beinhaltet einen Zwischenspeicher von dienst anbietenden Serviceinstanzen, deren Dienstmethoden durch sogenannte Reflektion, d.h. anhand von Informationen über das Ziel, den Namen und die Signatur der Methode ausgeführt werden. Zur Serialisierung und Deserialisierung der Parameter- und Rückgabeobjekte der Methodenaufrufe können XML-Dokument-Instanzen gemäss der SOAP-Spezifikation oder eine bandbreiteneffizientere Binärkodierung verwendet werden. Die Zuordnung von

klientenseitigen zu dienstanbieterseitigen Typen erfolgt im Falle der SOAP-(De)Serialisierung anhand von in den gegebenen Proxy-Service-Paaren deklarierten Typattributen, im Falle der Binärokodierung wird eine Bindung zwischen Serialisierung und zu instanziiierendem Typ gemäss den Anforderungen der vorgefundenen Deserialisierungskomponente implementiert.

Die Optimierung und Kontrolle der Kommunikation ist transparent sowohl für die Nutzer der angebotenen Dienste als auch für deren Anbieter.

Patentansprüche

1. System zur Kommunikation zwischen entfernten, bei Diensteanbietern (10) vorhandenen Objekten, deren Methoden als Webservices (5) zugreifbar sind, und Client-seitigen lokalen Stellvertretern (Proxies 3) in einem Rechnernetz (9), insbesondere Internet oder LAN, wobei

- a) auf der Seite der Diensteanbieter (10) zusätzlich zu vorhandenen Diensten (Webservices 5) ein Generalservice (8) installiert ist, der dafür eingerichtet ist, eine oder mehrere Serviceaufrufe eines Clients (1) an die vorhandenen Dienste (5) zu vermitteln, und eine oder mehrere Antwortnachrichten an den Client (1) zu übermitteln,
- b) Client-seitig zusätzlich zu den sonstigen lokalen Stellvertretern (Proxies 3) eine Optimierungsschicht (6) implementiert ist, die dafür eingerichtet ist, Client-seitige Optimierungen vorzunehmen und Aufrufbündel zusammenzustellen, und außerdem ein Generalproxy (7) installiert ist, der dafür eingerichtet ist, gebündelte Dienstaufträge auszuführen und Antwortnachrichten an die Optimierungsschicht (6) zurückzugeben.

2. System nach Anspruch 1, dadurch gekennzeichnet, dass die Optimierungsschicht (6) mindestens einen Zwischenspeicher enthält, mit dessen Hilfe Dienstaufträge vermeidbar oder verzögerbar sind.

3. System nach Anspruch 2, dadurch gekennzeichnet, dass der Client (1) mittels der Optimierungsschicht (6) und dem Generalproxy (7) dafür eingerichtet ist, auch ohne Aufruf durch eine Client-Anwendung (2) selbsttätig eine Kommunikation mit einem Diensteanbieter (10) zu initiieren, um gespeicherte Information zu aktualisieren.

4. System nach Anspruch 2, dadurch gekennzeichnet, dass der Client (1) mittels der Optimierungsschicht (6) und dem Generalproxy (7) dafür eingerichtet ist, zur Verwaltung, insbesondere Aktualisierung und Invalidierung der Daten im Zwischenspeicher, huckepack Informationen zusammen mit der Übertragung von Aufrufbündeln und der Rückübertragung von Antworten beim Dienstanbieter (10) anzufordern.

5. Verfahren zur Kommunikation zwischen entfernten, bei Dienstanbietern (10) vorhandenen Objekten, deren Methoden als Webservices (5) zugreifbar sind, und Client-seitigen lokalen Stellvertretern (Proxies 3) in einem Rechnernetz (9), insbesondere Internet oder LAN, wobei

- auf der Seite von Dienstanbietern (10) zusätzlich zu vorhandenen Diensten (Webservices 5) ein Generalservice (8) installiert ist, und
- Client-seitig jeweils zusätzlich zu den sonstigen lokalen Stellvertretern (Proxies 3) eine Optimierungsschicht (6) implementiert ist, die einen Zwischenspeicher enthält, und außerdem ein Generalproxy (7) installiert ist,

und wobei

- a) mehrere Aufrufe von Methoden durch Client-Anwendungen (2) beim jeweiligen Stellvertreter (Proxy 3) durch diesen der Optimierungsschicht (6) zugeleitet werden, dort zu einem Aufrufbündel zusammengestellt und einer Kommunikationsschicht (4) zugeleitet werden,
- b) eine Übertragung des Aufrufbündels zum Dienstanbieter (10) durchgeführt wird, dort durch den Generalservice (8) die im Aufrufbündel enthaltenen einzelnen Aufrufe den jeweils entsprechenden Diensten (5) zugeleitet werden, deren Antworten zusammengestellt und gebündelt zum Client (1) zurückübertragen werden, und
- c) die Antworten in der Optimierungsschicht (6) ausgewertet und über den jeweiligen Stellvertreter (3) der Client-Anwendung (2) zugeführt werden.

6. Verfahren nach Anspruch 5, dadurch gekennzeichnet, dass die Optimierungsschicht (6) zur Verwaltung, insbesondere Aktualisierung und

Invalidierung der Daten im Zwischenspeicher, huckepack Informationen zusammen mit der Übertragung von Aufrufbündeln und der Rückübertragung von Antworten beim Dienstanbieter (10) anfordert.

7. Verfahren nach Anspruch 5 oder 6, dadurch gekennzeichnet, dass die Optimierungsschicht (6) zur Verwaltung, insbesondere Aktualisierung und Invalidation der Daten im Zwischenspeicher, auch ohne Aufruf durch eine Client-Anwendung (2) selbsttätig eine Kommunikation mit einem Dienstanbieter (10) initiiert.

Zusammenfassung

System und Verfahren zur Kommunikation zwischen entfernten Objekten und lokalen Stellvertretern

Die Erfindung bezieht sich auf ein System – und ein entsprechendes Verfahren - zur Kommunikation zwischen entfernten, bei Dienstanibern (10) vorhandenen Objekten, deren Methoden als Webservices (5) zugreifbar sind, und Client-seitigen lokalen Stellvertretern (Proxies 3) in einem Rechnernetz (9), insbesondere Internet oder LAN. Auf der Seite der Dienstaniber (10) ist zusätzlich zu vorhandenen Diensten (Webservices 5) ein Generalservice (8) installiert, der dafür eingerichtet ist, eine oder mehrere Serviceaufrufe eines Clients (1) an die vorhandenen Dienste (5) zu vermitteln, und eine oder mehrere Antwortnachrichten an den Clients (1) zu übermitteln. Client-seitig ist jeweils zusätzlich zu den sonstigen lokalen Stellvertretern (Proxies 3) eine Optimierungsschicht (6) implementiert, die dafür eingerichtet ist, Client-seitige Optimierungen vorzunehmen und Aufrufbündel zusammenzustellen. Außerdem ist ein Generalproxy (7) installiert, der dafür eingerichtet ist, gebündelte Dienstaufträge auszuführen und Antwortnachrichten an die Optimierungsschicht (6) zurückzugeben.

Signifikante Fig.: Fig. 1

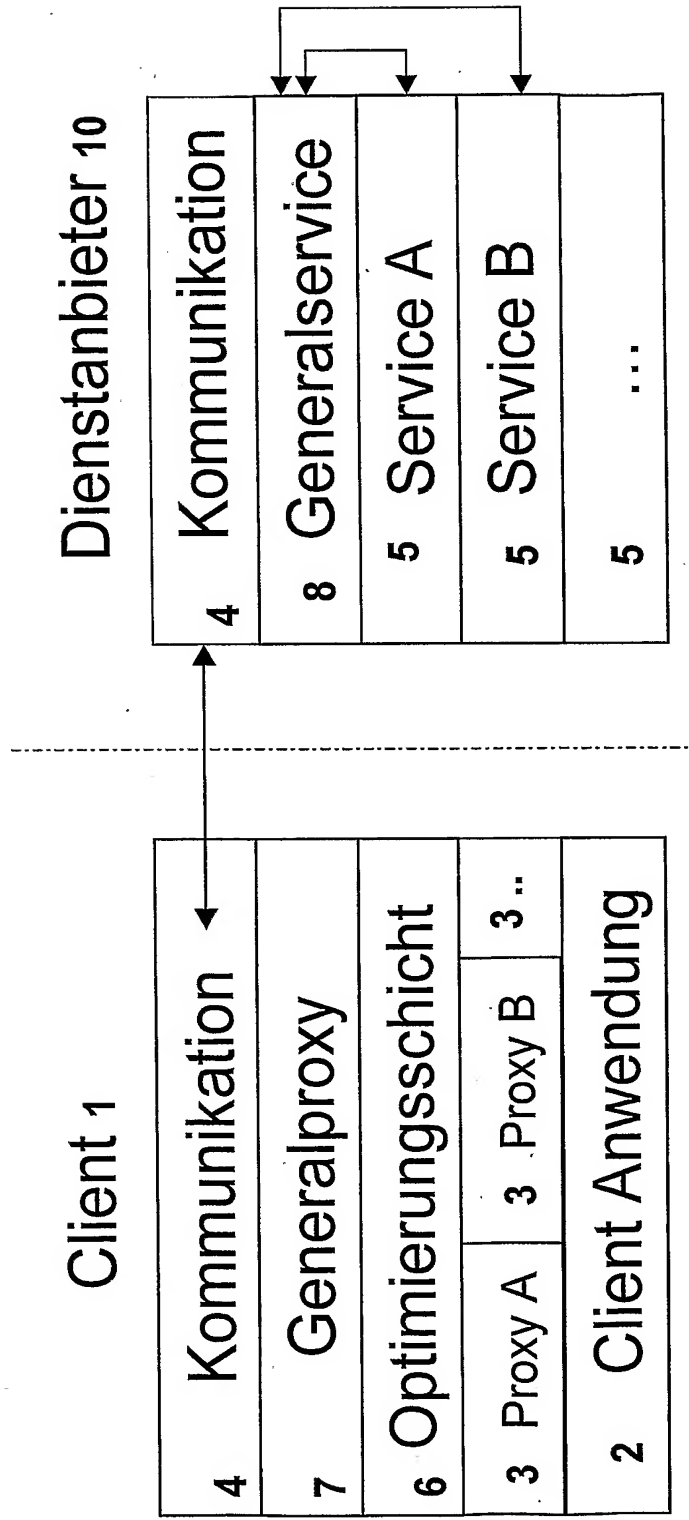


Fig. 1: Systemarchitektur Kommunikationssystem

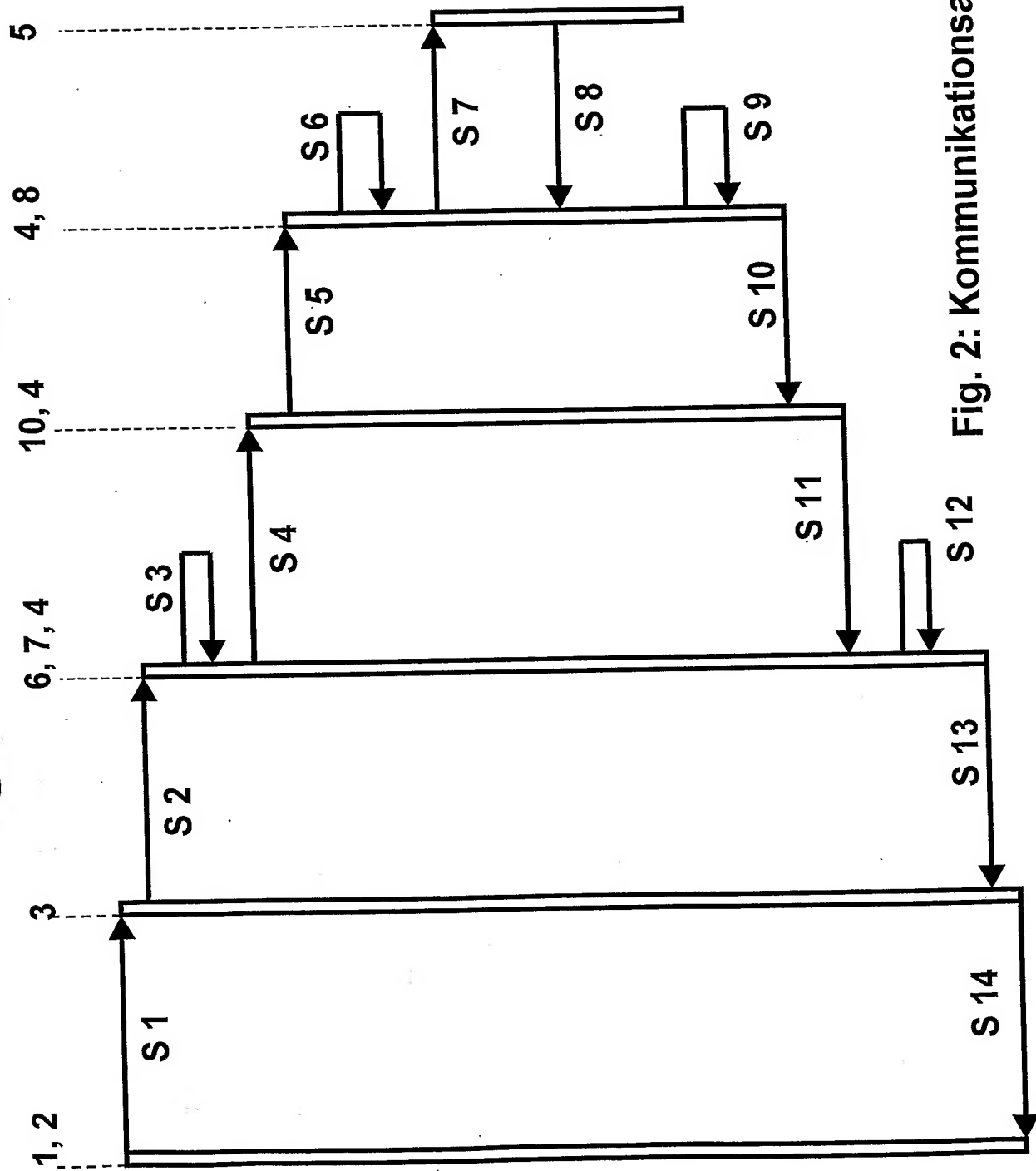


Fig. 2: Kommunikationsablauf

Internet/LAN 9

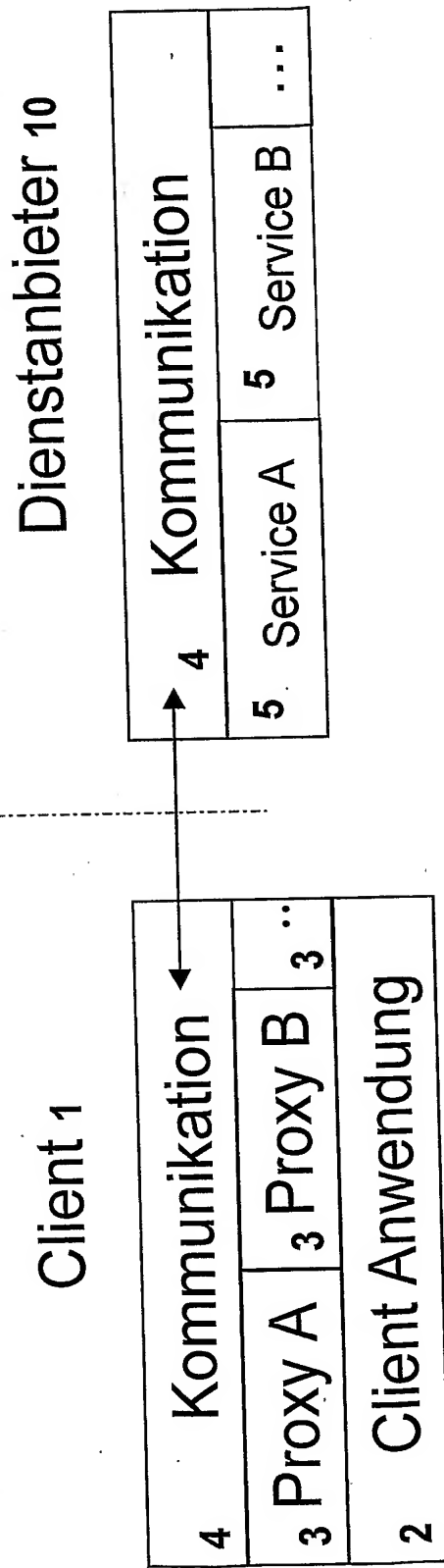


Fig. 3: Systemarchitektur – Stand der Technik